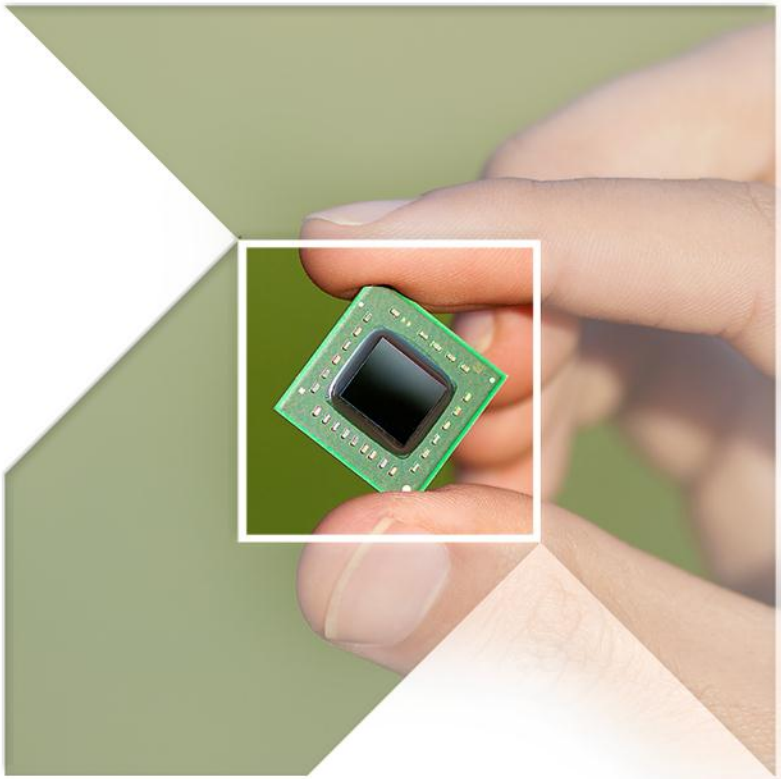


The Salishan Conference on
High Speed Computing

LANL / LLNL / SNL

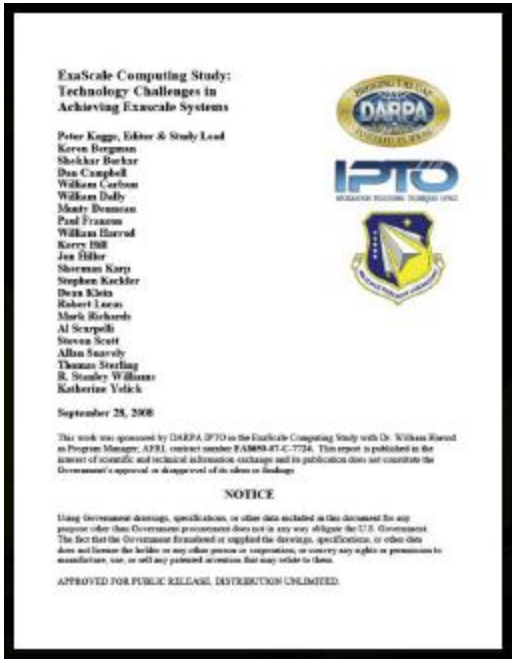
DATA PROCESSING IN EXASCALE-CLASS COMPUTER SYSTEMS

Chuck Moore
AMD Corporate Fellow &
Technology Group CTO

April 27, 2011

EXASCALE SYSTEM CHALLENGES

- DARPA study (2008) identified four major challenges:
  - Energy and Power challenge
  - Memory and Storage challenge
  - Concurrency and Locality challenge
  - Resiliency challenge
- All of these require deep consideration in the design of the compute nodes, the system-level fabric and the programming model
- Today's talk will focus on the *compute nodes and the associated programming model*


ExaScale Computing Study:
Technology Challenges in
Achieving Exascale Systems

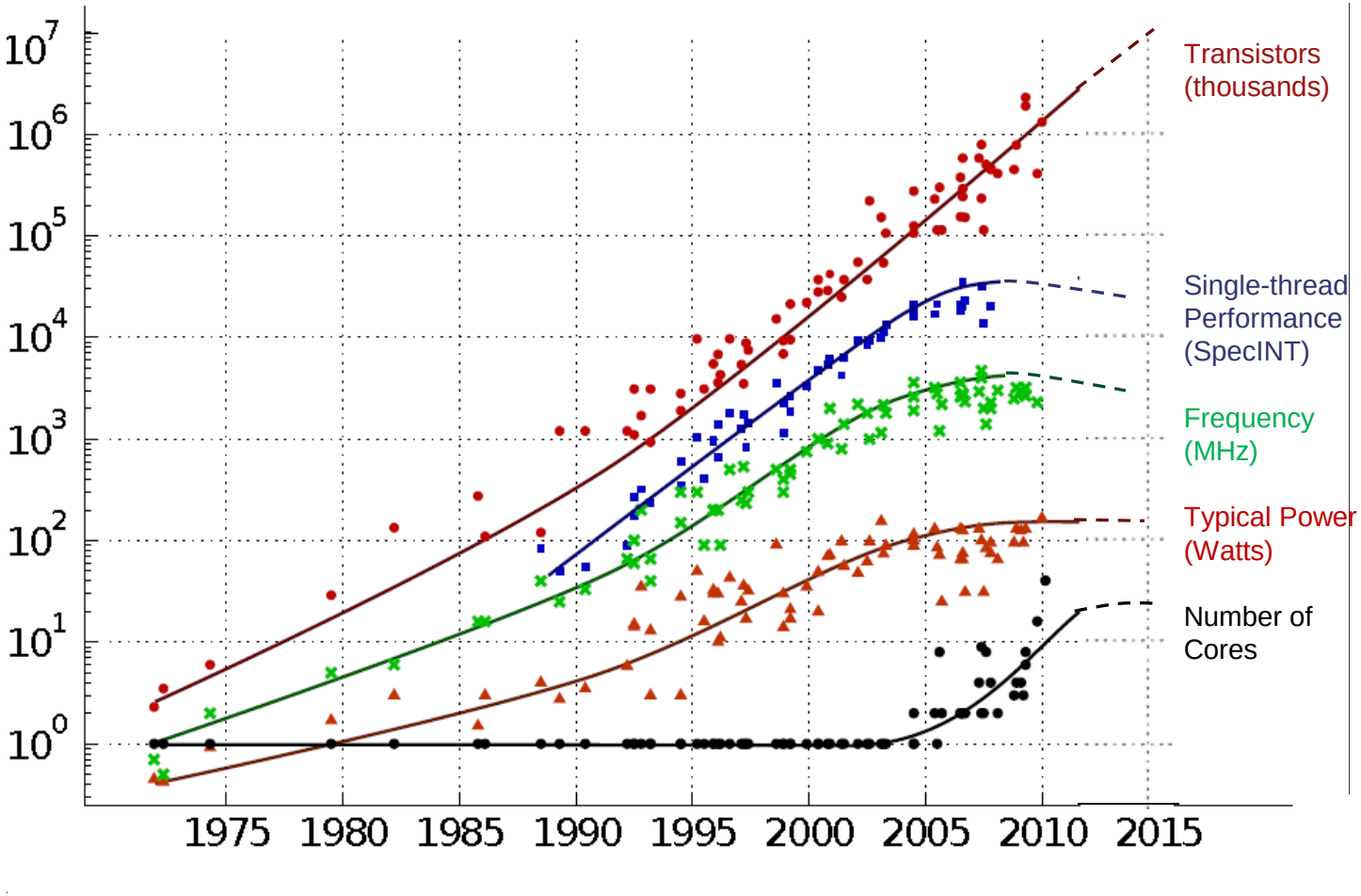
Peter Kogge, Editor & Study Lead
Keren Bergman
Shekhar Borkar
Dan Campbell
William Carlson
William Dally
Monty Denneau
Paul Franzon
William Harrod
Kerry Hill
Jon Hiller
Sherman Karp
Stephen Keckler
Dean Klein
Robert Lucas
Mark Richards
Al Scarpelli
Steven Scott
Allan Snavely
Thomas Sterling
R. Stanley Williams
Katherine Yelick

September 28, 2008

NOTICE

APPROVED FOR PUBLIC RELEASE, DISTRIBUTION UNLIMITED

35 YEARS OF MICROPROCESSOR TREND DATA



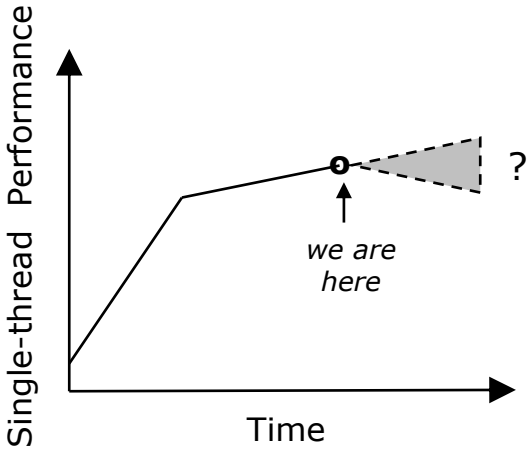
Original data collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond and C. Batten
Dotted line extrapolations by C. Moore

Three Eras of Processor Performance

Single-Core Era

Enabled by:
- ✓ Moore's Law
- ✓ Voltage Scaling
- ✓ MicroArchitecture

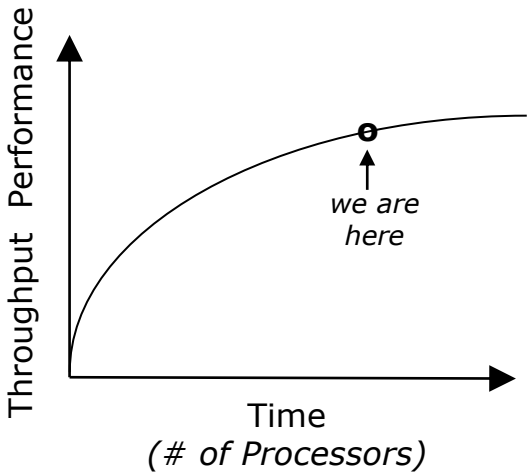
Constrained by:
- ✘ Power
- ✘ Complexity



Multi-Core Era

Enabled by:
- ✓ Moore's Law
- ✓ Desire for Throughput
- ✓ 20 years of SMP arch

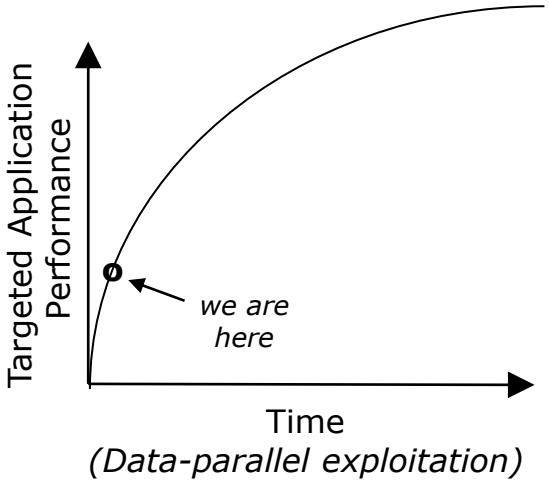
Constrained by:
- ✘ Power
- ✘ Parallel SW availability
- ✘ Scalability

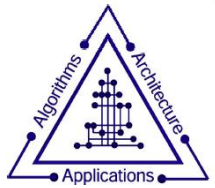


Heterogeneous Systems Era

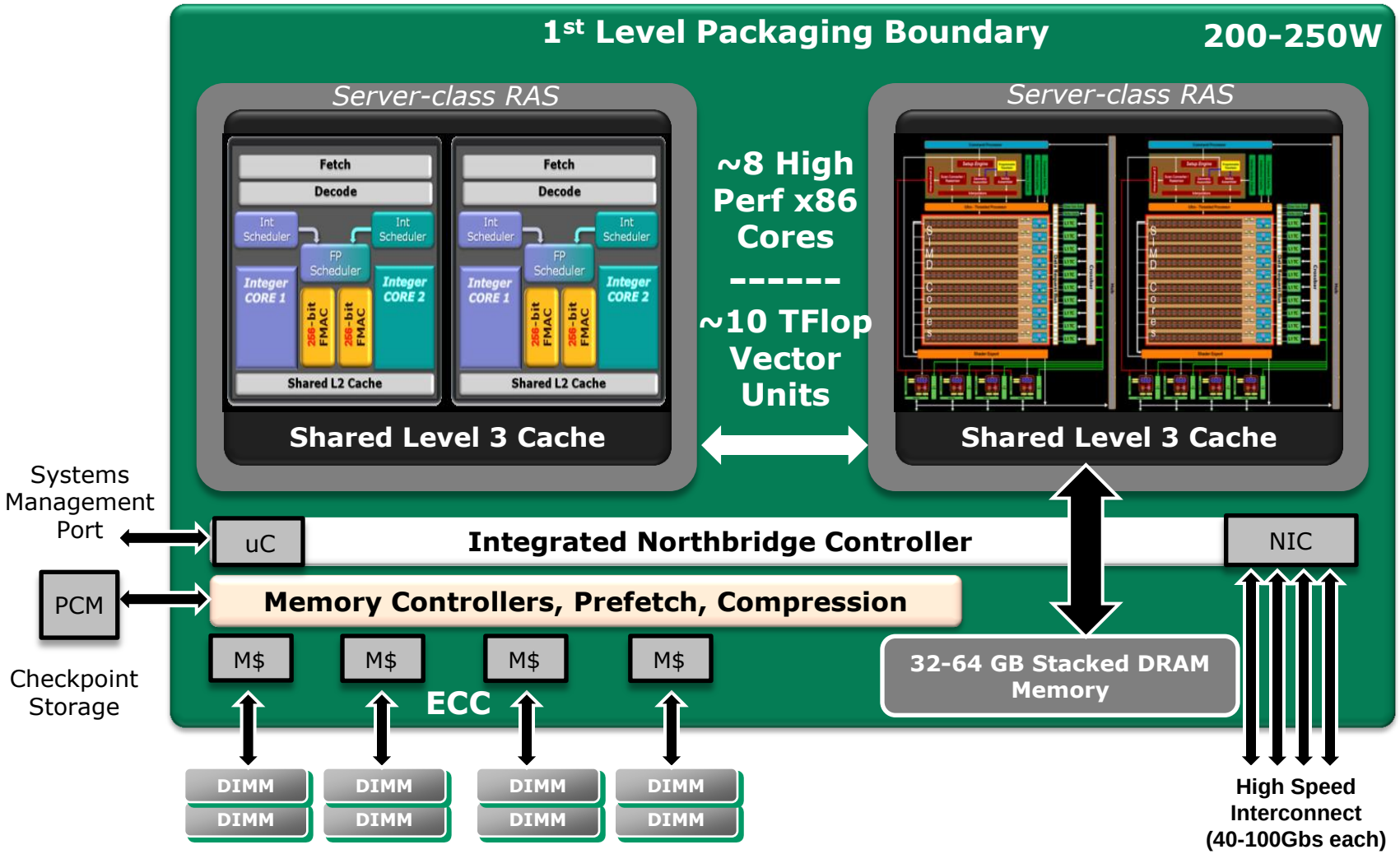
Enabled by:
- ✓ Moore's Law
- ✓ Abundant data parallelism
- ✓ Power efficient GPUs

Currently constrained by:
- ✘ Programming models
- ✘ Communication overheads





TECHNICALLY POSSIBLE HPC NODE FOR 2018

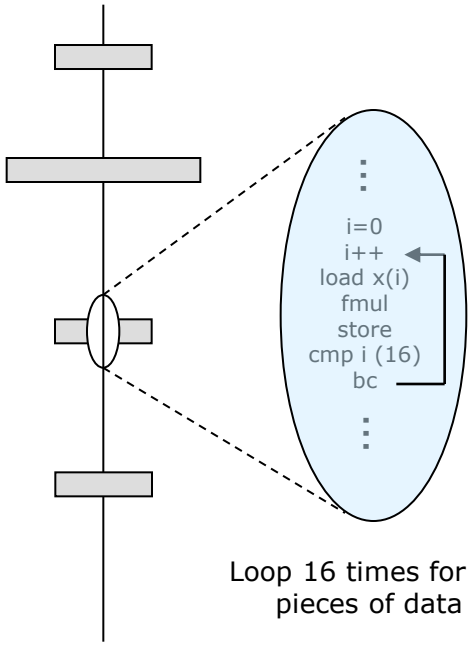


Concept represents engineering capability only, and is not intended as a product roadmap

Algorithms Architecture Applications

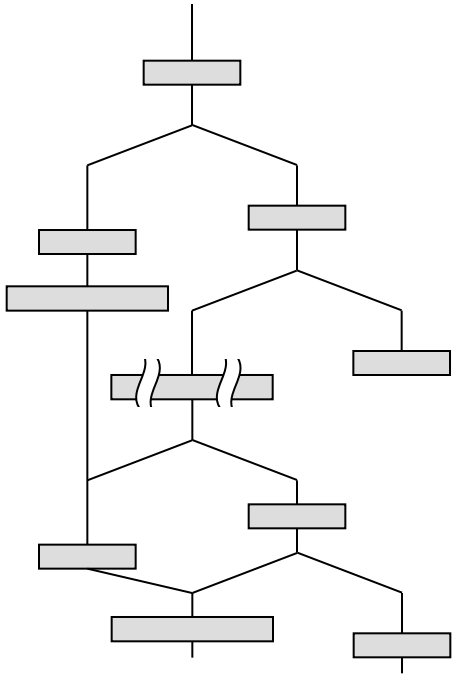
Challenge: *Nested Data Parallelism*

Fine-grain data parallel Code



Maps very well to integrated SIMD dataflow (SSE/AVX)

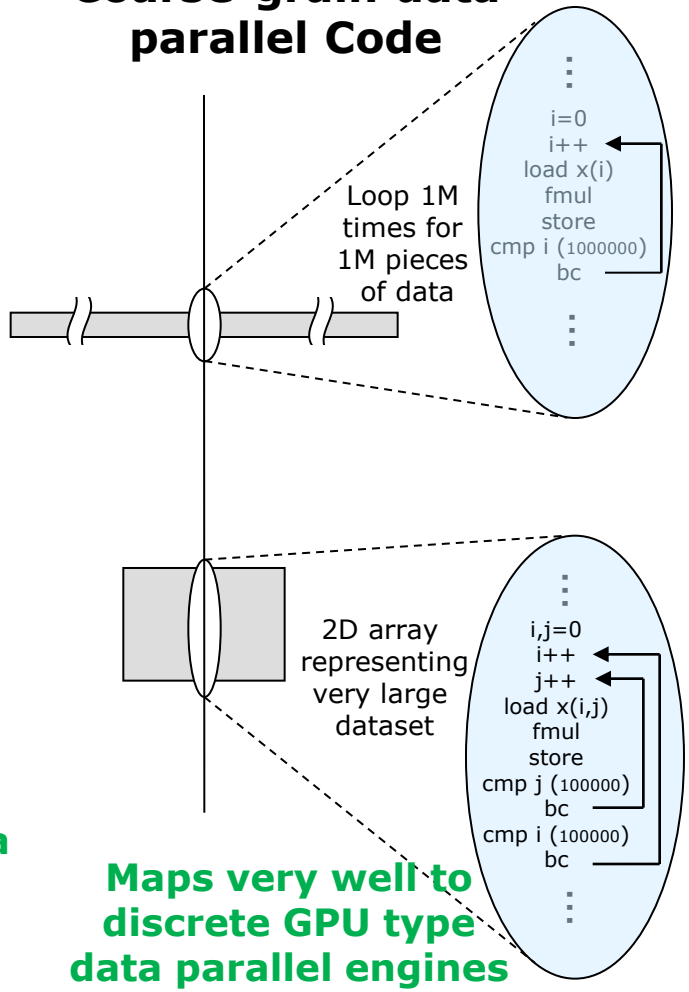
Nested data parallel Code



Lots of conditional data parallelism

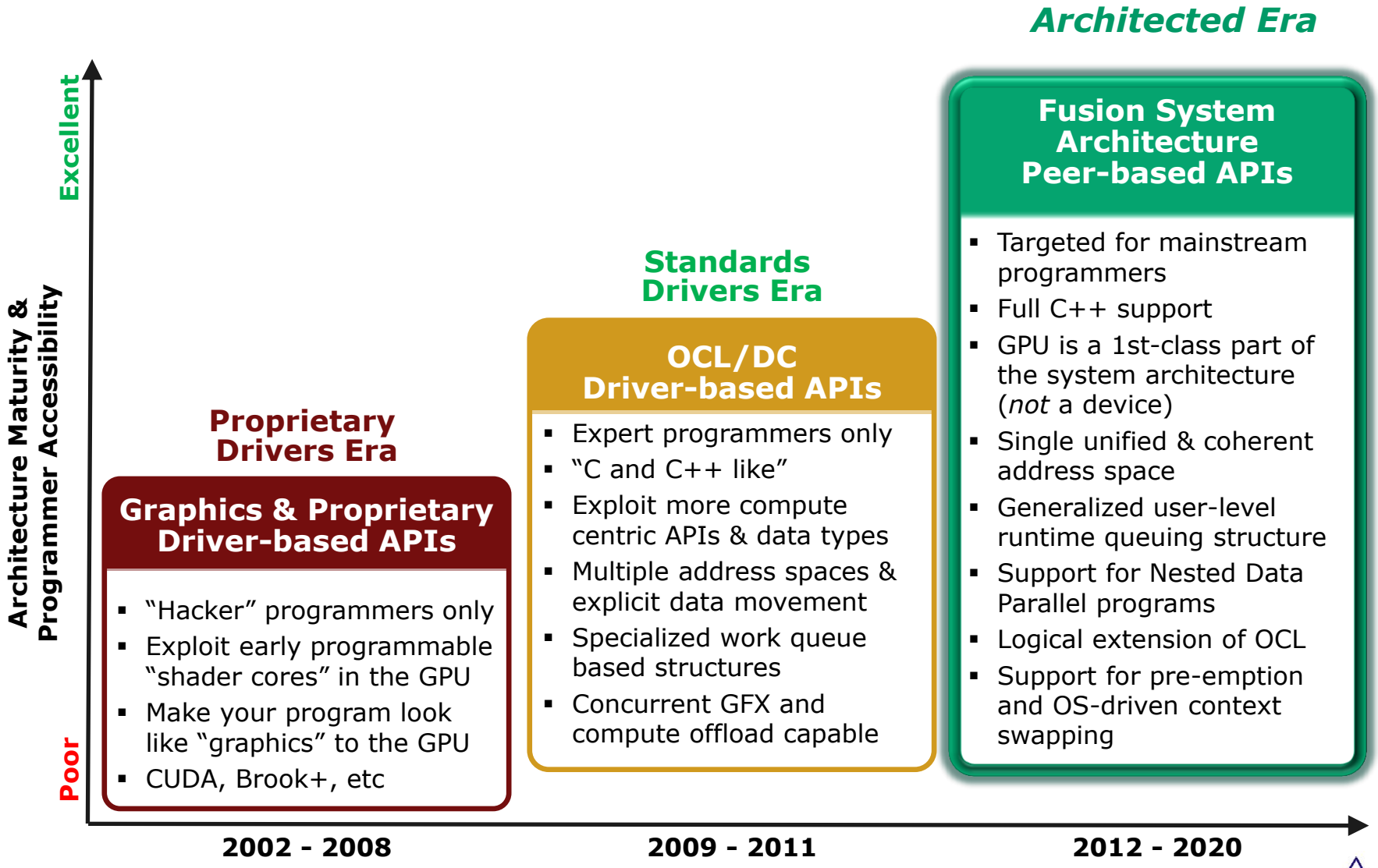
Benefits from closer coupling (CPU & GPU)

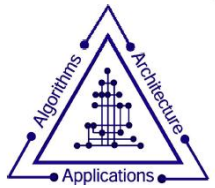
Coarse-grain data parallel Code



Maps very well to discrete GPU type data parallel engines

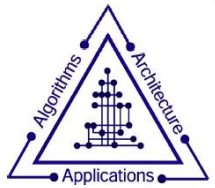
GPU COMPUTE OFFLOAD – 3 PHASES





WHAT IS THE FUSION SYSTEM ARCHITECTURE (FSA)?

- Foundation of AMD's heterogeneous compute platform strategy
- Enables data intensive applications to run on the most appropriate processor for the best possible performance and power
- Enables parallel task queuing runtimes to access the GPU directly and embrace heterogeneous computing
- Introduces an architected interface to drive the next platform standard for heterogeneous computing
- Continues to evolve and improve the standards based driver model for graphics, video and computing with discrete GPUs



MORE FSA DETAILS …

… WILL ALL BE RELEASED AT :

AMD's FUSION DEVELOPER SUMMIT

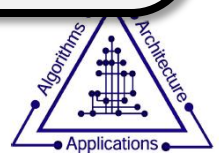
June 13-16, 2011 in Seattle, WA

- Execution Model
- Memory Model
- ABI Specifications
- Runtime Environment Specifications
- Programming Examples for OpenCL and DirectX11™
- And, much more …

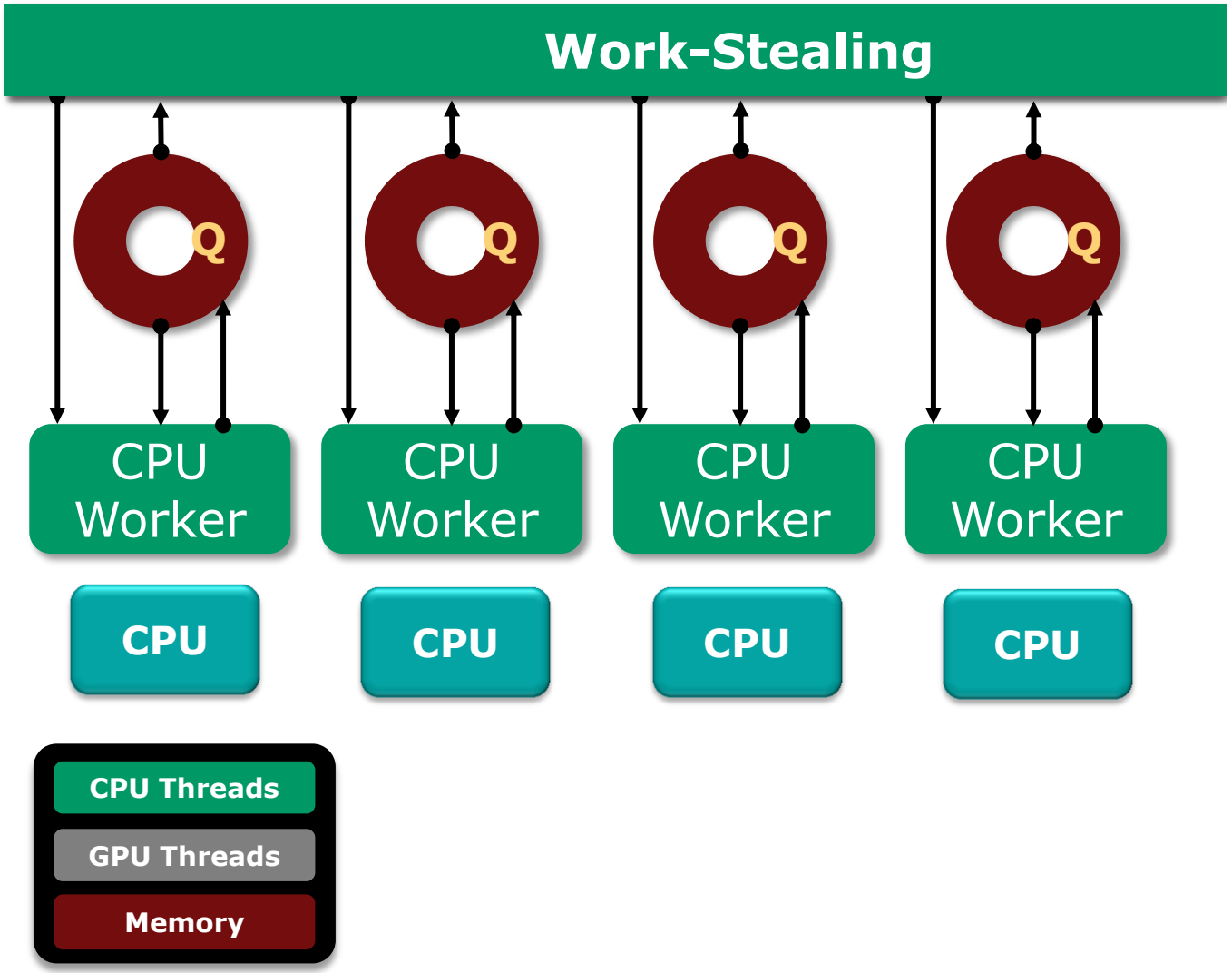
For more information: www.amd.com/afds

FSA EXECUTION MODEL OVERVIEW

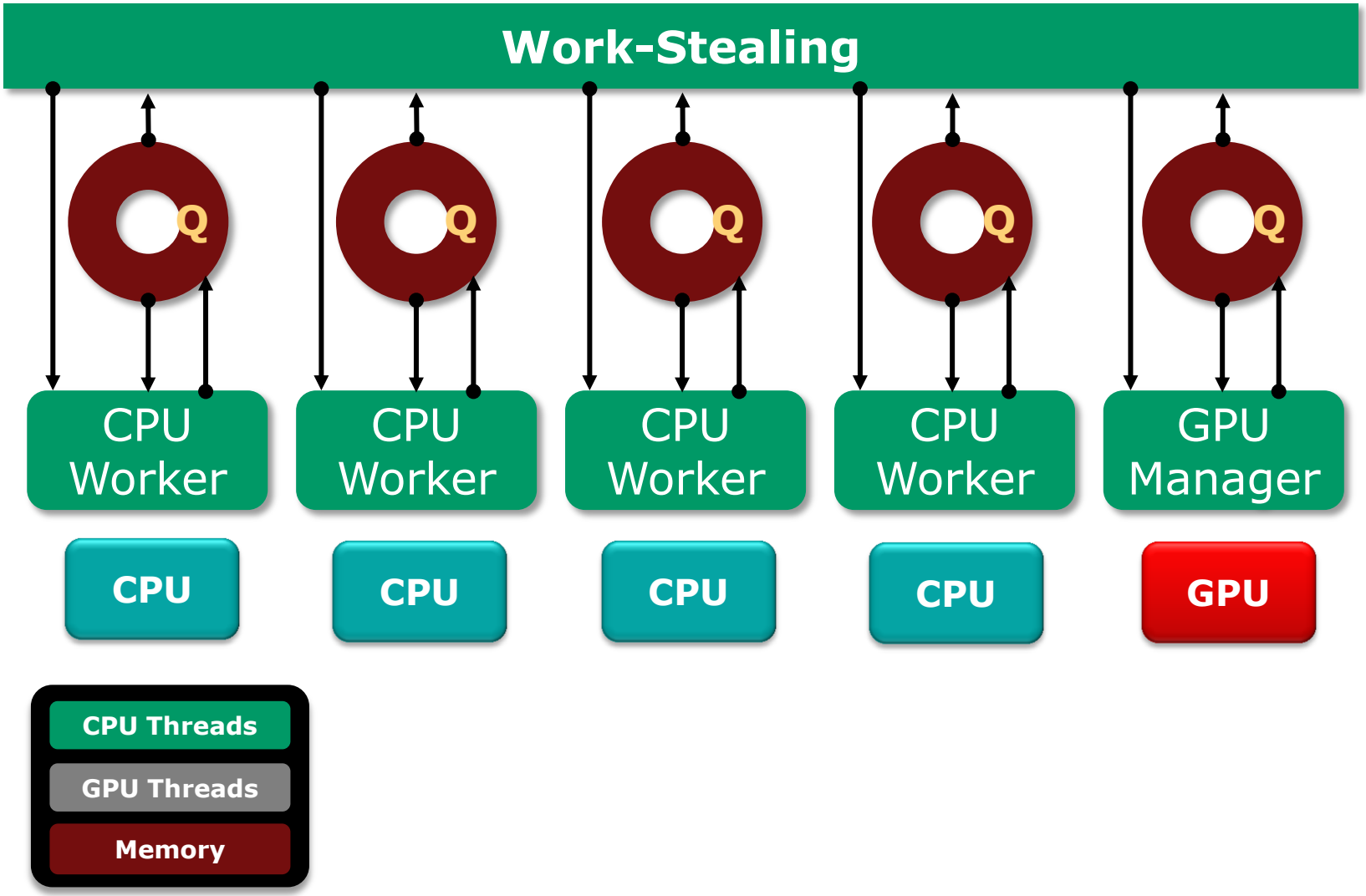
- Designed to support parallel task queuing runtimes
  - TBB, ConcRT, Grand Central
  - Allows programmers to manage tasks, not threads
  - Load balances through work stealing
- Architected Queuing Model
  - **CPU** -> **CPU** (already there)
  - **CPU** -> **GPU** (offload interface)
  - **GPU** -> **GPU** (recursion, tree traversal)
  - **GPU** -> **CPU** (syscalls, IO)
- GPU support for function pointers, recursion, exceptions, etc
- User mode queuing
  - Low latency dispatch directly from applications
  - GPU hardware processes queues from multiple processes in round robin manner
  - Progress notifications written directly to memory and employ standard OS signaling
  - No new CPU instructions required



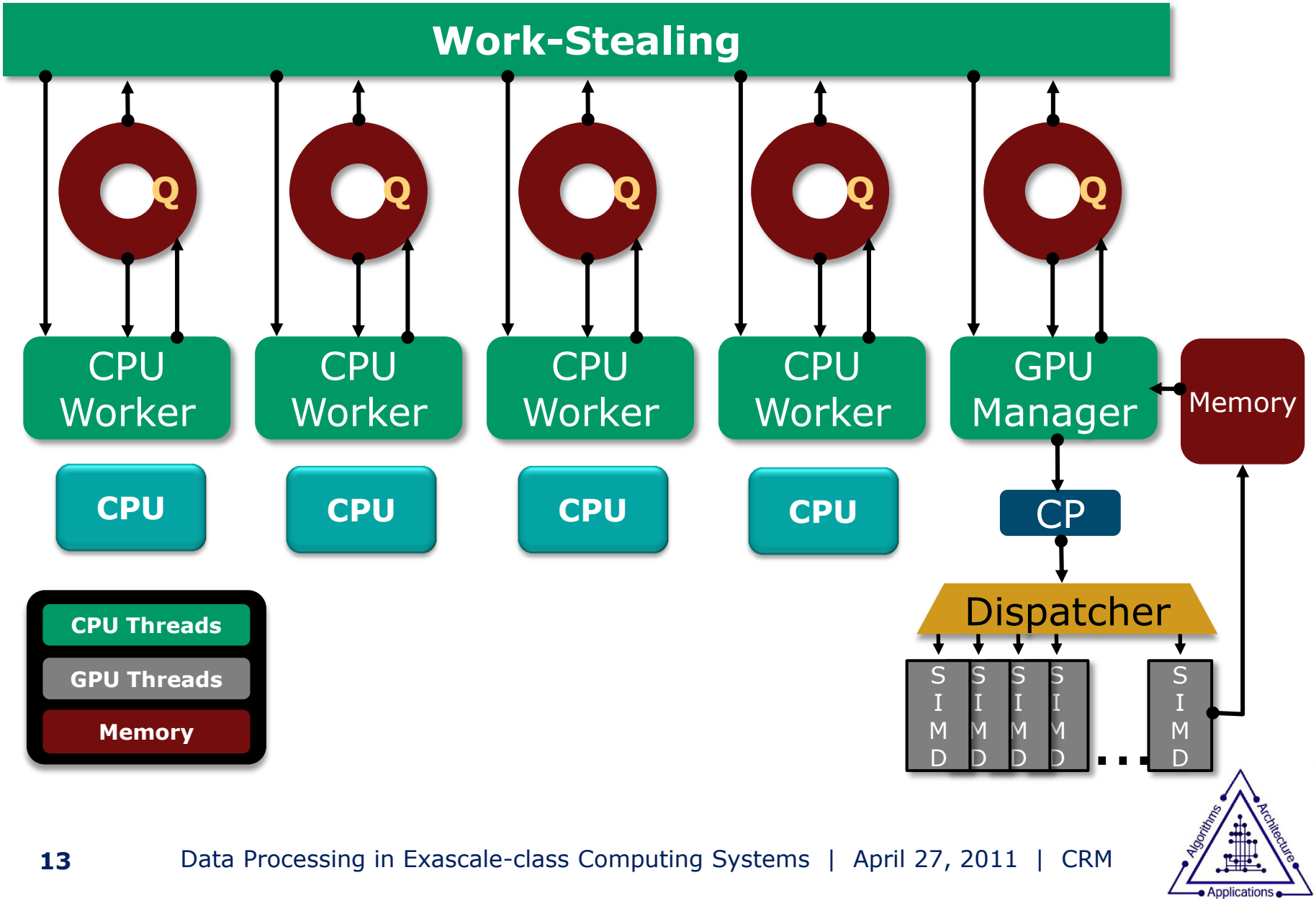
TASK QUEUING RUNTIME ON CPUS



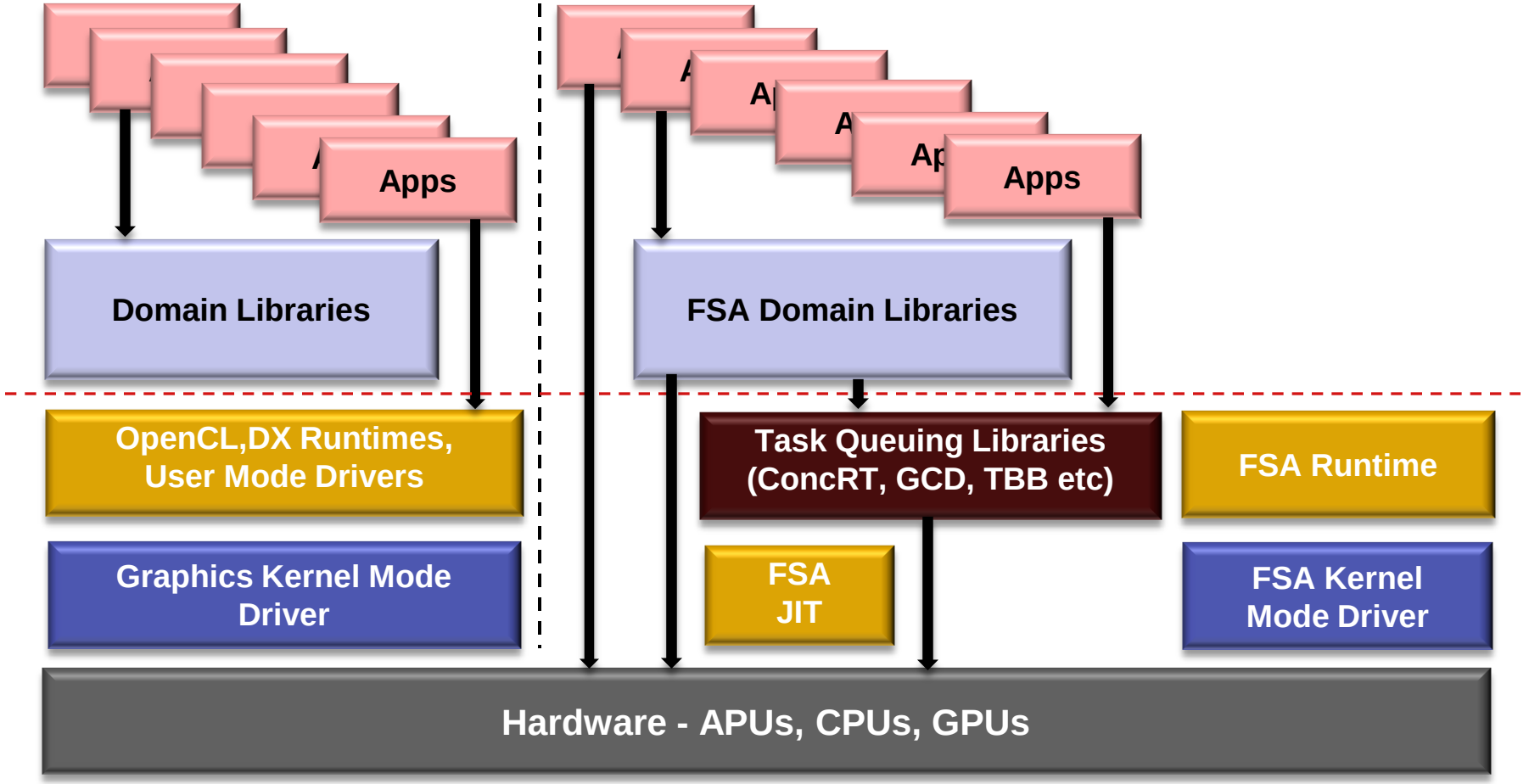
TASK QUEUING RUNTIME ON THE FSA PLATFORM



TASK QUEUING RUNTIME ON THE FSA PLATFORM



DRIVER AND FSA SOFTWARE STACKS



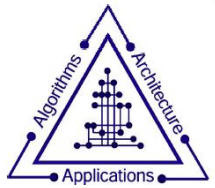
AMD user mode component

AMD kernel mode component

All others contributed by third parties or AMD

PROGRAMMING AT SCALE (100K NODES!)

- Reality: Different hardware technologies present different characteristics in the full system design
  - NUMA – Latency, BW, coherency support
  - Networking Tiers – Variable communication, synch, data movement costs
  - Reliability
- Some of these can be flattened out with brute force increased cost
  - Not always practical or desirable to do so …
- The balance of these differences are presented to SW
  - ***Glass half empty***: Programmability, performance or both suffer ☹
  - ***Glass half full***: Fast time-to-functionality; then, gradual optimization opportunity as needed ☺
- Abstractions that present these realities in simple ways are needed:
  - Locality
  - Optimized Communications: PGAS extended addressing
  - RAS support and non-disruptive checkpointing
  - Resource and systems management @ scale (system hypervisor)

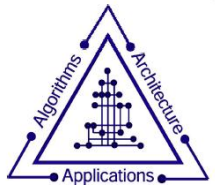


COMMONALITIES BETWEEN MEGA-DATACENTERS AND SUPERCOMPUTERS?

Attribute	Mega-Datacenter	Supercomputer	Commonality?
Scale	10K → 100K nodes	10K → 100K nodes	Maybe
Workload diversity	Moderate, but almost all ***integer*** processing	Moderate, but almost all ***floating point*** processing	No
Top optimization priorities	1. TCO (CapEx and OpEx) 2. Throughput/W/$$	1. Sustained FLOPs 2. TCO	Yes -- TCO
Node-level power	10's of Watts	100's of Watts	No -- Power Density
Memory BW & Cap	High	Very High	Yes
Interconnect BW	Working towards 100GbE	Working towards 1 Tbps	Maybe
Interconnect Topology	Embrace Tiers w/ natural parallelism boundaries	Desire as Flat as possible	No -- Costs
Programming model	Exploit natural parallelism to build fast response time Map/Reduce, Hadoop	MPI, OpenMP, OpenCL (!)	Maybe
Power Provisioning	Close proximity to cheap power	Close proximity to electric power generation facilities	Yes
Cooling	Sophisticated heat flow engineering	Sophisticated heat flow engineering	Yes
Dealing w/ Failures	Fail-in-place; self recovering algorithms	Working towards extensive checkpointing	Maybe

SUMMARY AND CONCLUDING REMARKS

- 10+TFLOP processing nodes for Exascale-class Supercomputers are within reach by 2018
  - TSV Stacked Memory → Reasonable Memory Bandwidth (0.1B/FLOP)
  - 15nm, 11nm, 8nm CMOS Roadmap → CPU & Large GPU integration
  - Balance → Leading edge x86 sequential and GPU/Vector machines
- Full system architectures, such as AMD's Fusion System Architecture, will help make GPU/Vector programming approachable by more people
  - Unified system address space → A pointer is a pointer
  - X86-based page tables throughout → User-level Task Parallel Runtimes
  - Reduced overhead of offload → Finer granularity offload
- Modern Mega-datacenters share some of the same problems as the upcoming class of Supercomputers
  - The MDC guys have HUGE financial motivations for continuing to mature this technology
  - Alignment is well advised ☺



DISCLAIMER

The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions and typographical errors.

AMD MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREOF AND ASSUMES NO RESPONSIBILITY FOR ANY INACCURACIES, ERRORS OR OMISSIONS THAT MAY APPEAR IN THIS INFORMATION.

AMD SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT WILL AMD BE LIABLE TO ANY PERSON FOR ANY DIRECT, INDIRECT, SPECIAL OR OTHER CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF ANY INFORMATION CONTAINED HEREIN, EVEN IF AMD IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Trademark Attribution

AMD, the AMD Arrow logo and combinations thereof are trademarks of Advanced Micro Devices, Inc. in the United States and/or other jurisdictions. Other names used in this presentation are for identification purposes only and may be trademarks of their respective owners.